

ПОСТРОЕНИЕ 3D-МОДЕЛЕЙ ОБЪЕКТОВ С ПОМОЩЬЮ OPENGL

Чеботарева Е.Н., Аксёнов С.В.
Томский политехнический университет
enc1@tpu.ru

Введение

В последнее время все большую популярность приобретают объемные изображения объектов, их трехмерные модели, что находит широкое применение в робототехнике, науке, медицине, коммерческой деятельности, а также в системах виртуальной реальности и в других сферах деятельности.

Поэтому возникла потребность в автоматизированных системах построения трехмерных моделей по изображениям, не требующих дорогостоящей дополнительной аппаратуры, в которых взаимодействие с пользователем сводится к малому количеству простых операций.

Краткий обзор технологий и библиотек

Наиболее популярными технологиями для работы с графикой, в том числе и трехмерной, являются OpenGL и DirectX.

DirectX (от англ. direct — прямой, непосредственный) — это набор API, разработанных для решения задач, связанных с программированием под Microsoft Windows. Наиболее широко используется при написании компьютерных игр. [1]

OpenGL (Open Graphics Library) — спецификация, определяющая независимый от языка программирования платформонезависимый программный интерфейс для написания приложений, использующих двумерную и трехмерную компьютерную графику.

Включает более 300 функций для рисования сложных трехмерных сцен из простых примитивов. Используется при создании компьютерных игр, САПР, виртуальной реальности, визуализации в научных исследованиях. [2]

Из указанного выше следует, что наиболее перспективным является использование OpenGL, так как данная спецификация не зависит от языка программирования и платформы, и кроме того, не является проприетарной. Необходимо использовать современную версию данной графической библиотеки, начиная с OpenGL 3.3.

В качестве графической библиотеки был выбран freeglut, благодаря своей популярности, открытости и стабильности.

freetglut — открытая альтернатива OpenGL Utility Toolkit (GLUT). GLUT (и, следовательно, freetglut) позволяет пользователю создавать окна, предоставляющие контекст OpenGL на широком спектре платформ, и управлять ими, а также взаимодействовать с мышью, клавиатурой и джойстиком.

Трехмерная фигура, полученная с использованием freetglut приведена на рис. 1.

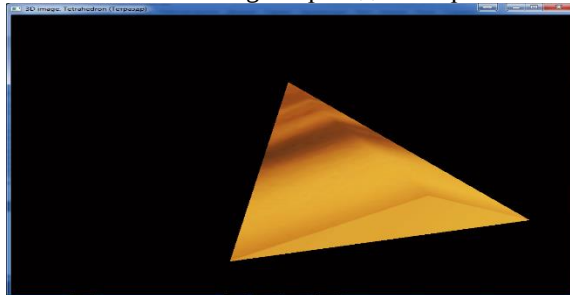


Рис.1. Пример трехмерной фигуры в OpenGL

С того времени, как оригинальный GLUT прекратил развитие, freetglut начал развиваться с целью улучшения предоставляемого инструментария.

freetglut сейчас очень стабилен и имеет меньше ошибок, чем оригинальный GLUT. [3]

Для того, чтобы определить, что поддерживает видеокарта (рис. 2) и избежать ошибок, существует специальная библиотека GLEW. [6] В данном случае она выполняет вспомогательную роль.

OpenGL Extension Wrangler Library (GLEW) - кроссплатформенная библиотека на C/C++, которая упрощает запрос и загрузку расширений OpenGL. GLEW обеспечивает эффективные run-time механизмы для определения того, какие OpenGL расширения поддерживаются на целевой платформе. Все расширения OpenGL размещаются в одном заголовочном файле, который автоматически генерируется из официального списка расширений. GLEW доступна на множестве операционных систем, включая Windows, Linux, Mac OS X, FreeBSD, IRIX и Solaris. [4]

```
Some technical information.  
Driver supports OpenGL 3.3  
Details:  
Using GLEW 1.13.0  
Vendor: ATI Technologies Inc.  
Renderer: AMD Radeon(TM) HD 6520G  
Version: 4.2.11665 Compatibility Profile Context  
GLSL: 4.20
```

Рис.2. Информация о видеокarte и поддержке графических библиотек, полученная с помощью GLEW

В качестве языка программирования был выбран C++, так он совместим с OpenGL и вышеуказанными библиотеками и является кроссплатформенным.

C++ — компилируемый, статически типизированный язык программирования общего назначения.

Поддерживает такие парадигмы программирования как процедурное

программирование, объектно-ориентированное программирование, обобщённое программирование, обеспечивает модульность, раздельную компиляцию, обработку исключений, абстракцию данных, объявление типов (классов) объектов, виртуальные функции.

C++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также развлекательных приложений (игр). [5]

Алгоритм построения трехмерных объектов

Необходимо осуществлять построение 3D-моделей объектов с помощью OpenGL для отображения результатов, полученных из расчетов на основе двумерных изображений. При построении должно учитываться расстояние до объекта, чтобы объекты находящиеся на разном расстоянии от камеры, окрашивались в цвет соответствующий этой дистанции. Таким образом, должны получиться разноцветные слои. Результат должен быть выведен на экран средствами OpenGL в виде окрашенных трехмерных объектов.

Для осуществления этого необходимо использовать 2 потока: первый поток – расстояние до объекта (каждому значению расстояния соответствует свой цвет), второй поток – сами изображения этого конкретного объекта.

Далее необходимо:

произвести наложение изображения на матрицу расстояний (окраска).

Если изображение содержит несколько объектов, необходимо выделить области, принадлежащие каждому объекту соответствующей текстурой, так как они могут быть на одном расстоянии и, значит окрашены в одинаковые цвета и поэтому неразличимы на полученном изображении. Текстура помогает решить эту проблему.

Осуществить соединение граней объекта, так как на каждом из изображений объекта видны разные его грани.

Если изображения объекта будут получены из видео, необходимо получить кадры. Поэтому необходимо будет подобрать подходящую частоту выборки кадров, которые затем рассматриваются как последовательности простых двумерных изображений данного объекта.

Произвести отрисовку полученной трехмерной модели объекта средствами OpenGL.

При решении данной задачи OpenGL предоставляет средства, позволяющие эффективно и без дополнительных затрат памяти и времени на написание кода. Фигуры разделяются на составляющие треугольники и их вершины,

имеющие координаты. Для их использования создается буфер вершин (GLuint VBO), который привязывается к данным. Индексная отрисовка позволяет сэкономить ресурсы и избавиться от много раз повторяющихся координат вершин. Для этого создается массив индексов, которые хранятся в соответствующем буфере. [6, 7]

Вершинный и фрагментный (пиксельный) шейдеры позволяют удобно хранить и использовать данные, которые не изменяются в ходе выполнения программы. [6]

Для работы с координатами, преобразованиями, камерой и текстурой были созданы соответствующие классы, использующие элементы OpenGL и реализующие различные математические функции и преобразования.

Использование буфера глубины позволяет избавиться от дефектов при изображении, в случае, когда объект находится близко к камере. [7]

Заключение

При решении задачи построение трехмерных моделей объектов оптимальным является использование языка программирования C++ и библиотеки OpenGL, так как они совместимы с технологией CUDA, позволяющей осуществлять параллельные вычисления на графическом процессоре, что позволит достичь более высокой производительности.

Литература

1. Википедия – DirectX [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/DirectX>, свободный. (Дата обращения: 20.09.2015 г.)
2. Википедия – OpenGL [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/OpenGL>, свободный. (Дата обращения: 20.09.2015 г.)
3. Википедия – Freeglut [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/Freeglut>, свободный. (Дата обращения: 20.09.2015 г.)
4. Википедия – OpenGL Extension Wrangler Library [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/OpenGL_Extension_Wrangler_Library, свободный. (Дата обращения: 20.09.2015 г.)
5. Википедия – C++ [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/C%2B%2B>, свободный. (Дата обращения: 20.09.2015 г.)
6. OpenGL Development Cookbook / Muhammad Mubeen Movania – Birmingham – Mumbai, Pack Publishing, 2013. – 311 p.
7. Tutorials for modern OpenGL (3.3+) [Электронный ресурс]. Режим доступа: <http://www.opengl-tutorial.org/>, свободный. (Дата обращения: 20.09.2015 г.)